

AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (currently amended) A processor, comprising:

at least one interface engine adapted to be connected to at least one external device located externally of the processor; and

a programmable pipeline adapted to directionally transfer data packets through the pipeline from a first end of the pipeline to a second end of the pipeline, and adapted to perform sequences of instructions on the data packets,

the pipeline comprising plural access points located in a spaced-apart relation between the first end of the pipeline and the second end of the pipeline,

wherein the at least one interface engine is connected to each of the plural access points, [[and]]

wherein the at least one interface engine is adapted

i) to receive a request from any one of the plural access points of the programmable pipeline, the request being received upon arrival of one of the data packets at the respective any one access point,

ii) to send a request output to the external device, the request output being based at least partly on the request from the one access point,

iii) responsive to the request output, to receive an external reply from the external device, and

iv) to send to the pipeline a response, based on the external reply, to the request, and

wherein the interface engine includes an arbiter configured to allow forwarding of requests from the plural access points in a fair manner between each of the plural access points.

2. (previously presented) A processor according to claim 1, whereby

the request comprises a first request code, according to a first coding scheme, the interface engine being adapted to execute a program, the execution being dependent upon the first request code, and to obtain, as a result of the execution of the program, at least one device control code, according to a second coding scheme, and

one of i) the interface engine is adapted to send the device control code to the external device, and ii) the request output sent by the interface engine is based at least partly on the device control code.

3. (original) A processor according to claim 2, whereby the device control code is an operational code of the external device.

4. (previously presented) A processor according to claim 2, whereby the program is stored in a microcode memory included in the interface engine.

5. (previously presented) A processor according to claim 1, the interface engine comprising a reply control unit adapted to receive at least one receiver ID signal related to the request, and to determine, based on the receiver ID signal, the access point which is to receive the response.

6. (previously presented) A processor according to claim 5, whereby the reply control unit is adapted to receive an input control signal, and timing information for receiving the external reply from the external device is determined from the input control signal.

7. (previously presented) A processor according to claim 1, whereby the number of access points adapted to send a request to the interface engine is adjustable.

8. (currently amended) A method in a processor comprising a programmable pipeline and at least one interface engine, adapted to be connected to at least one external device located externally of the processor, the processor being adapted to directionally transfer data packets from a first end of the pipeline to a second end of the pipeline, the pipeline comprising a plurality of access points located in a spaced-apart relation between the first end of the pipeline and the second end of the pipeline, the at least one interface engine being connected to each of the ~~plural~~ plurality of access points, the interface engine comprising an arbiter configured to allow forwarding of requests from the plurality of access points in a fair manner between each of the plurality of access points, the method comprising the steps of:

receiving a request from any one of the access points of the programmable pipeline, the request being received upon arrival of a data packet at the respective any one access point,

sending a request output to the external device, the request output based at least partly on the request from the one access point,

responsive to the request output, receiving an external reply from the external device, and

sending to the pipeline a response, based on the external reply, to the request.

9. (previously presented) A method according to claim 8, wherein

the request comprises a first request code, according to a first coding scheme, the method further comprising the step of executing a program, the execution being dependent upon the first request code, to obtain at least one device control code, according to a second coding scheme, and

one of i) the device control code is sent to the external device, and ii) the request output sent by the interface engine is based at least partly on the device control code.

10. (previously presented) A method according to claim 9, whereby the device control code is an operational code of the external device.

11. (previously presented) A method according to claim 9, whereby the program is stored in a microcode memory included in the interface engine.

12. (currently amended) A method according to claim 8, further comprising the steps of:

[[-]] sending at least one receiver ID signal, related to the request, to a re-play control unit included in the interface engine, and

[[-]] determining, based on the receiver ID signal, the access point which is to receive the response.

13. (previously presented) A method according to claim 12, further comprising the step of sending to the reply control unit an input control signal, wherein timing information for receiving the external reply from the external device is determined from the input control signal.

14. (previously presented) A method according to claim 8, whereby the number of access points adapted to send a request to the interface engine is adjustable.

15. (new) A processor according to claim 1, fair manner of the arbiter is a round-robin manner between the plurality of access points.

16. (new) A method according to claim 8, wherein the fair manner of the arbiter is a round-robin manner between the plurality of access points.

17. (new) A processor, comprising:

a programmable pipeline with a plurality of access points located in a spaced-apart relation between a first end of the pipeline and a second end of the pipeline, the pipeline

configured to directionally transfer data packets from the first end to the second end; and

an interface engine, having a reply control unit, connected to an external device located externally of the processor, and connected to each of the plurality of access points, the interface engine configured to

i) receive a request from the pipeline upon arrival of a data packet at any one of the plurality of access points,

ii) send a request output to the external device, the request output based at least partly on the request,

iii) responsive to the request output, receive an external reply from the external device, and

iv) send a response, based on the external reply, to the pipeline,

and the reply control unit configured to

i) receive a receiver ID signal related to the request, and

ii) determine, based on the receiver ID signal, the any one of the plurality of access points of the pipeline to receive the response.

18. (new) A processor according to claim 17, wherein a quantity of the plurality of access points from which the request is received by the interface engine from the pipeline is adjustable based on one of

- i) a data flow rate through the pipeline, and
- ii) a flow capacity of the external device.

19. (new) A processor according to claim 17, wherein the interface engine includes an arbiter configured to allow forwarding of requests from the plurality of access points in a fair manner between each of the plurality of access points.

20. (new) A method according to claim 19, wherein the fair manner of the arbiter is a round-robin manner between the plurality of access points.